

# Infrastructure Automation using Terraform on Azure

**Publish Custom Module in Terraform Registry**



# Table of Contents

Prerequisite:.....	3
Walkthrough: .....	3
Part 1: Login to your GitHub Account .....	3
Part 2: Create a new repository .....	8
Part 3: Upload files.....	10
Part 4: Create a new release .....	12
Part 5: Publish the Module in Terraform Registry .....	15

## Infrastructure Automation using Terraform on Azure – Lab Guide

This Activity demonstrates how to publish custom modules in Terraform Registry. By publishing modules, we can share our code with others.

### Prerequisite:

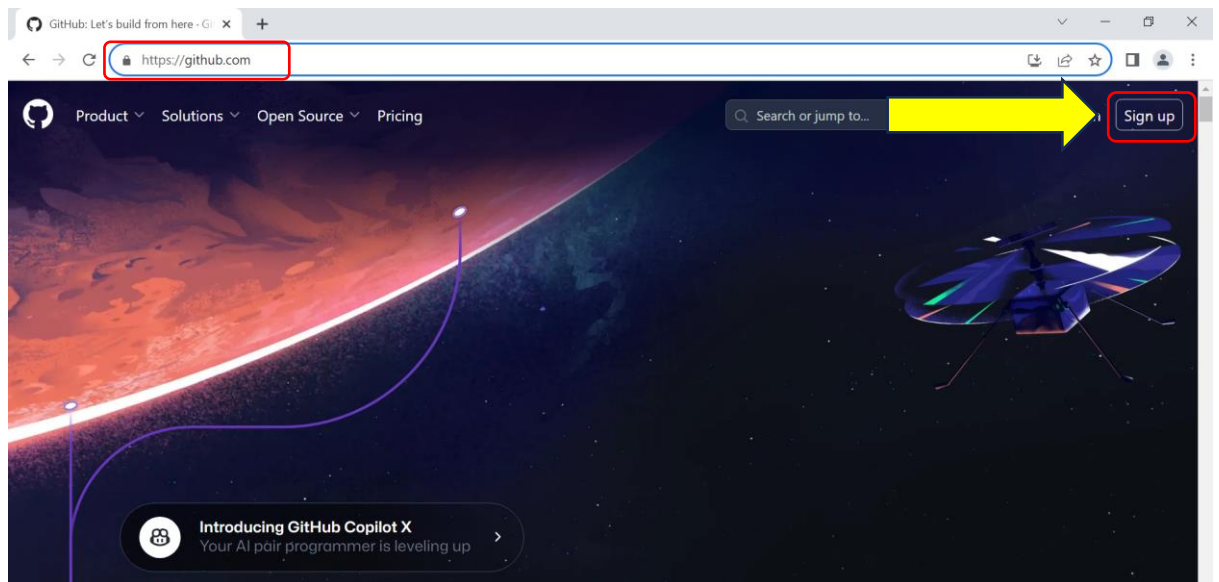
- 1) Need GitHub Account

### Walkthrough:

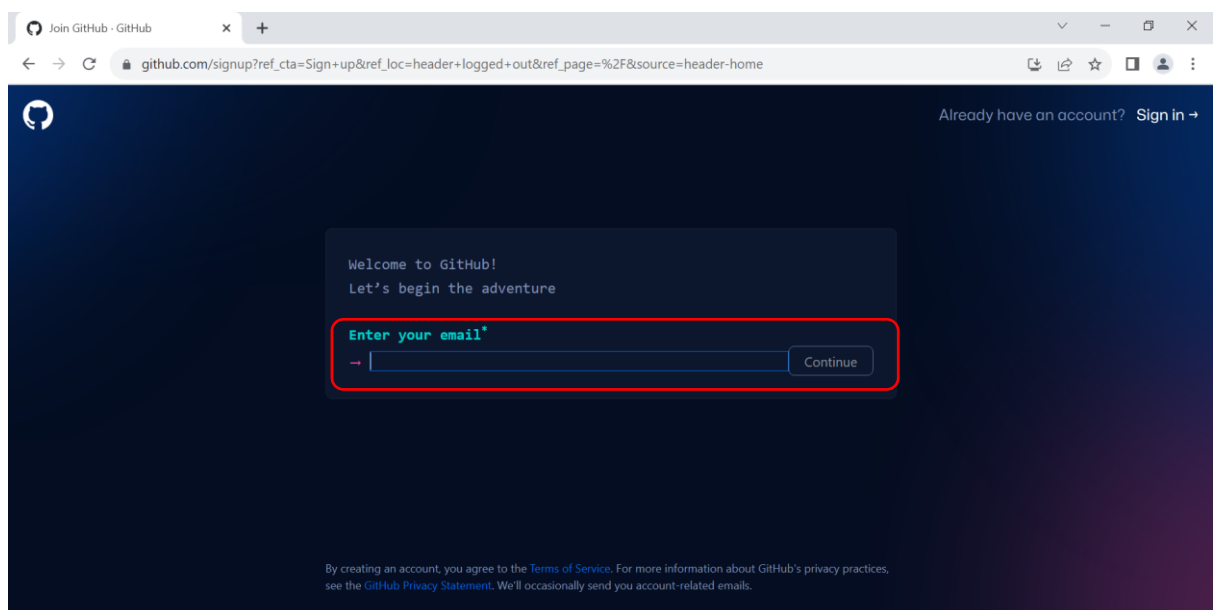
1. Login to your GitHub Account
2. Create a new repository
3. Upload files
4. Create a new release
5. Publish the Module in Terraform Registry

### Part 1: Login to your GitHub Account

- 1 If you already have GitHub Account, Sign-In using your username and password. Otherwise please follow below steps to create a GitHub Account. Navigate to <https://github.com/> and click on Sign-Up.

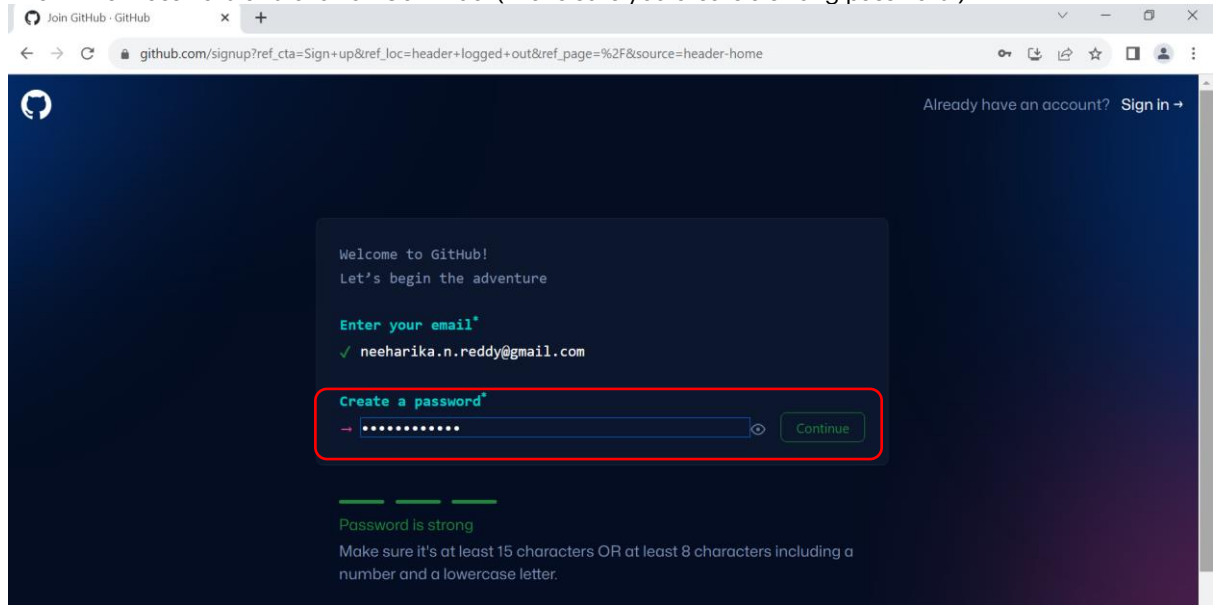


- 2 Provide your email-id and click on continue.



3

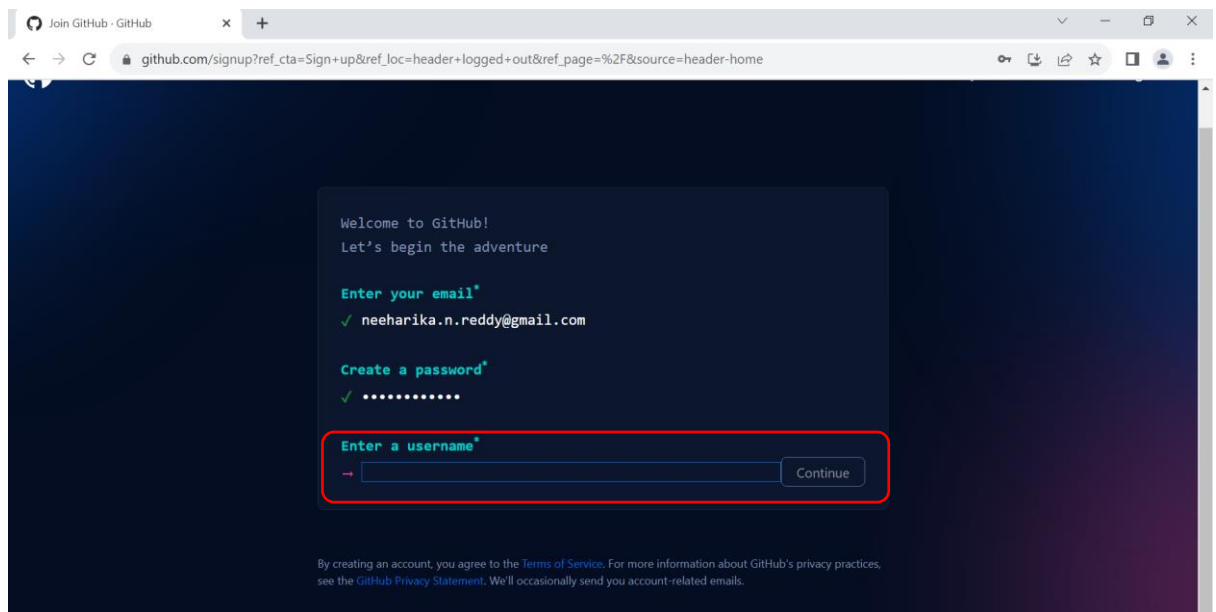
Then Enter Password and click on Continue. ( Make sure you create a strong password ).



The screenshot shows the GitHub signup page. The browser address bar displays 'github.com/signup?ref\_cta=Sign+up&ref\_loc=header+logged+out&ref\_page=%2F&source=header-home'. The page content includes the GitHub logo, a 'Sign in' link for existing users, and a welcome message. The 'Enter your email' field is filled with 'neeharika.n.reddy@gmail.com'. The 'Create a password' field is highlighted with a red box and contains a strong password. A 'Continue' button is next to it. Below the password field, a green message states 'Password is strong' and provides additional requirements: 'Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter.'

4

Next enter Username and click on continue. If you want product updates and announcements, you can opt-in for it. Otherwise just type "n" and click on continue.

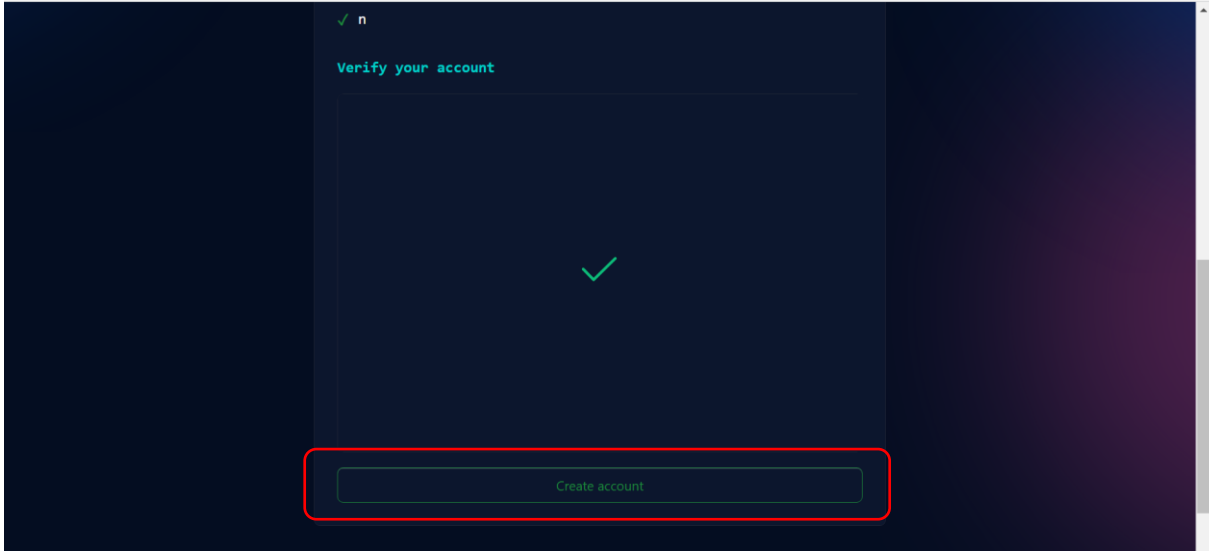
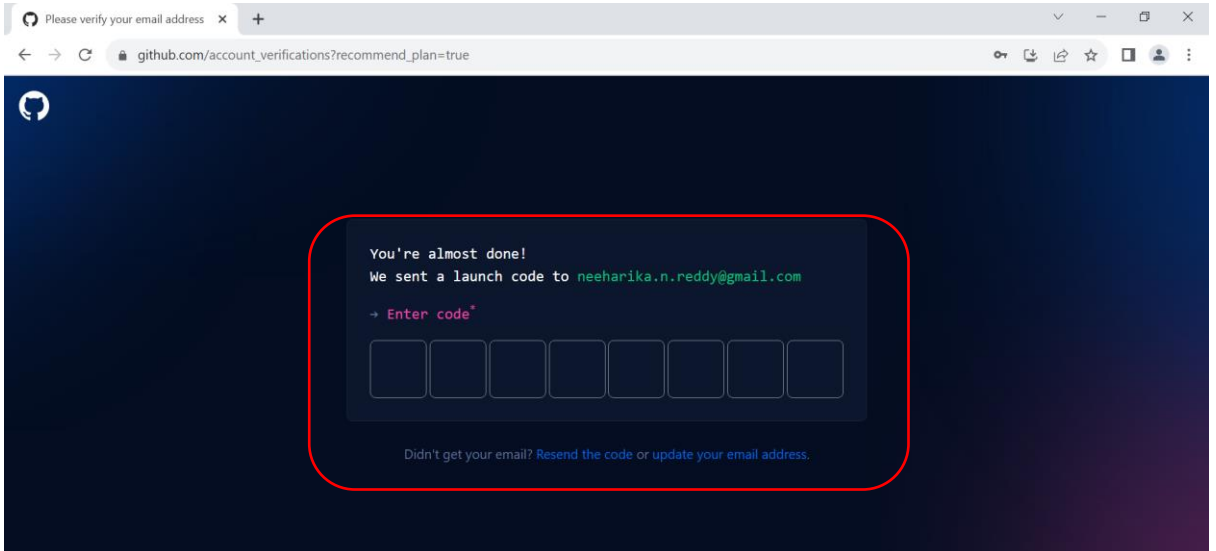


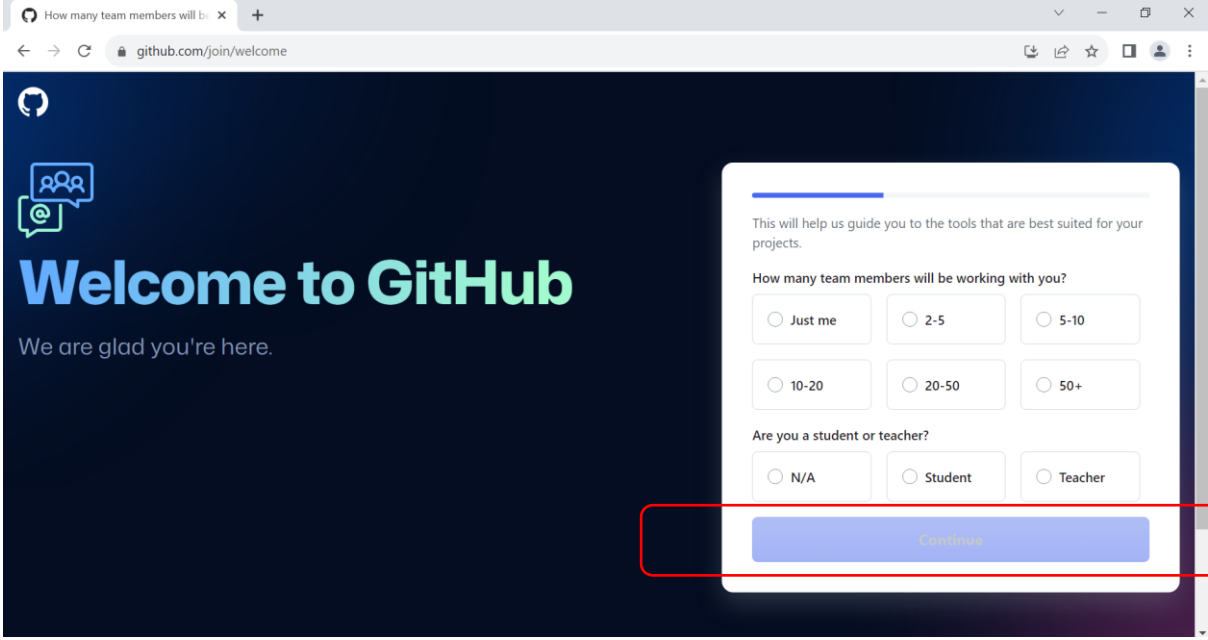
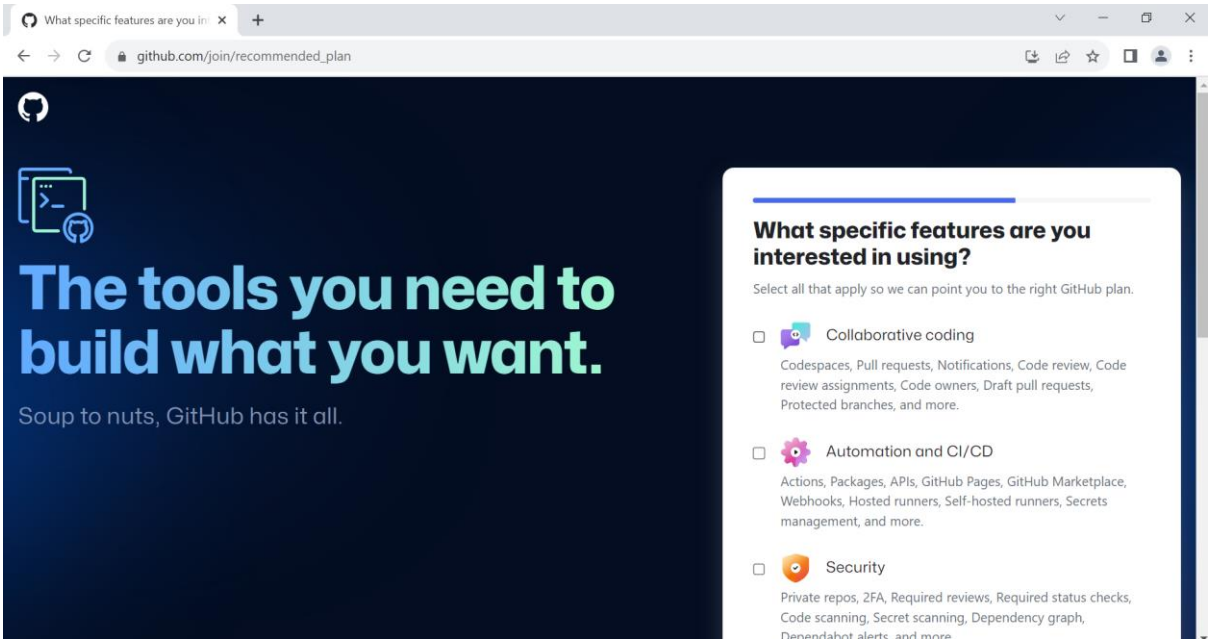
The screenshot shows the GitHub signup page at the 'Enter a username' step. The browser address bar is the same. The 'Enter your email' and 'Create a password' fields are now filled and marked with green checkmarks. The 'Enter a username' field is highlighted with a red box and contains the letter 'n'. A 'Continue' button is next to it. At the bottom, a small text block states: 'By creating an account, you agree to the Terms of Service. For more information about GitHub's privacy practices, see the GitHub Privacy Statement. We'll occasionally send you account-related emails.'

## Infrastructure Automation using Terraform on Azure – Lab Guide

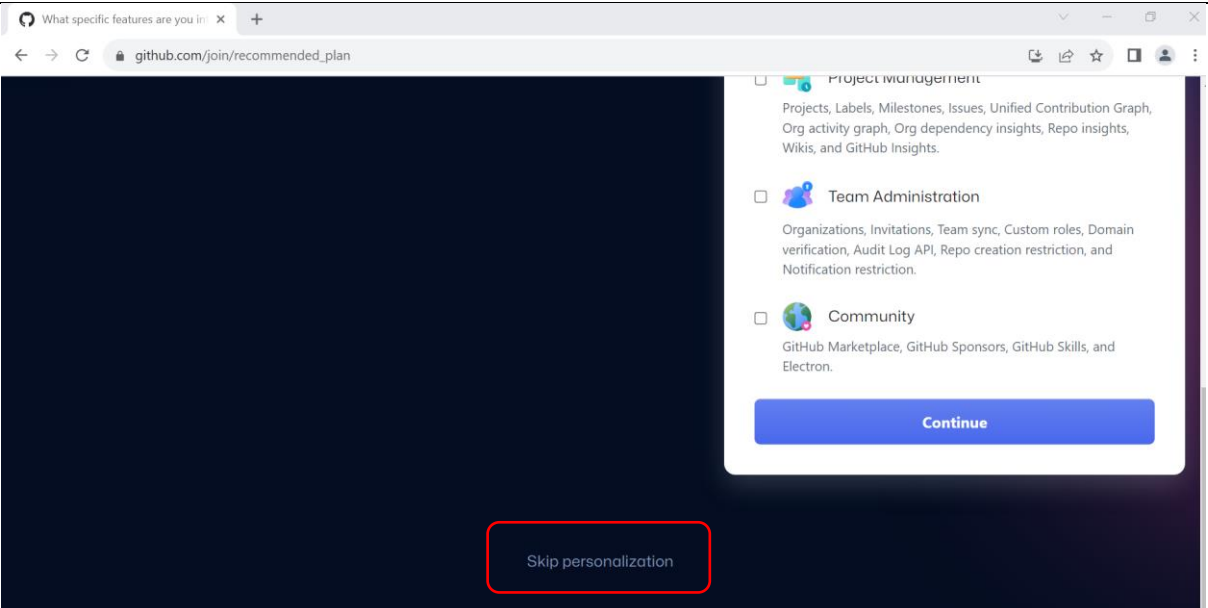
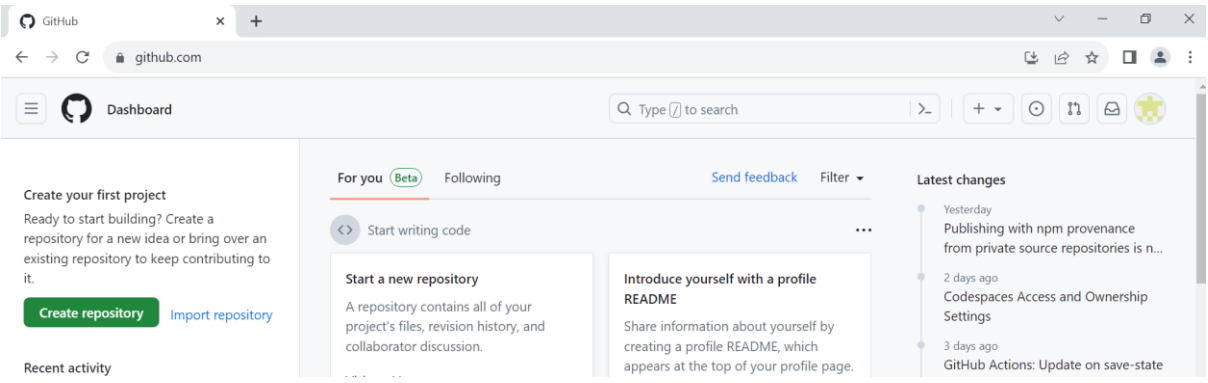
5	<p>Verify your account by solving the puzzle and then click on “Create Account”.</p>

## Infrastructure Automation using Terraform on Azure – Lab Guide

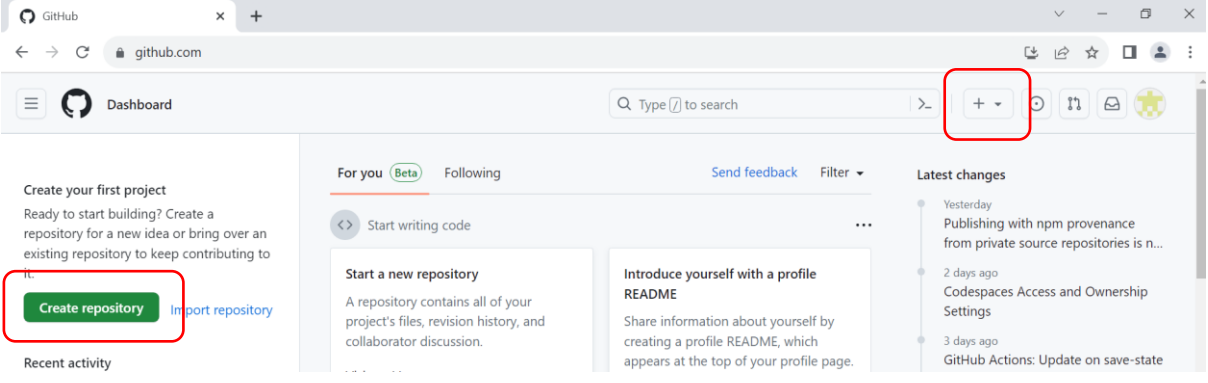
	
6	<p>Enter the code which is sent to your email-id. After verification, it will ask you to provide your preferences. Specify your preferences and Click on Continue.</p> 

	
7	<p>Scroll down to the bottom and click on “Skip Personalization”.</p> 

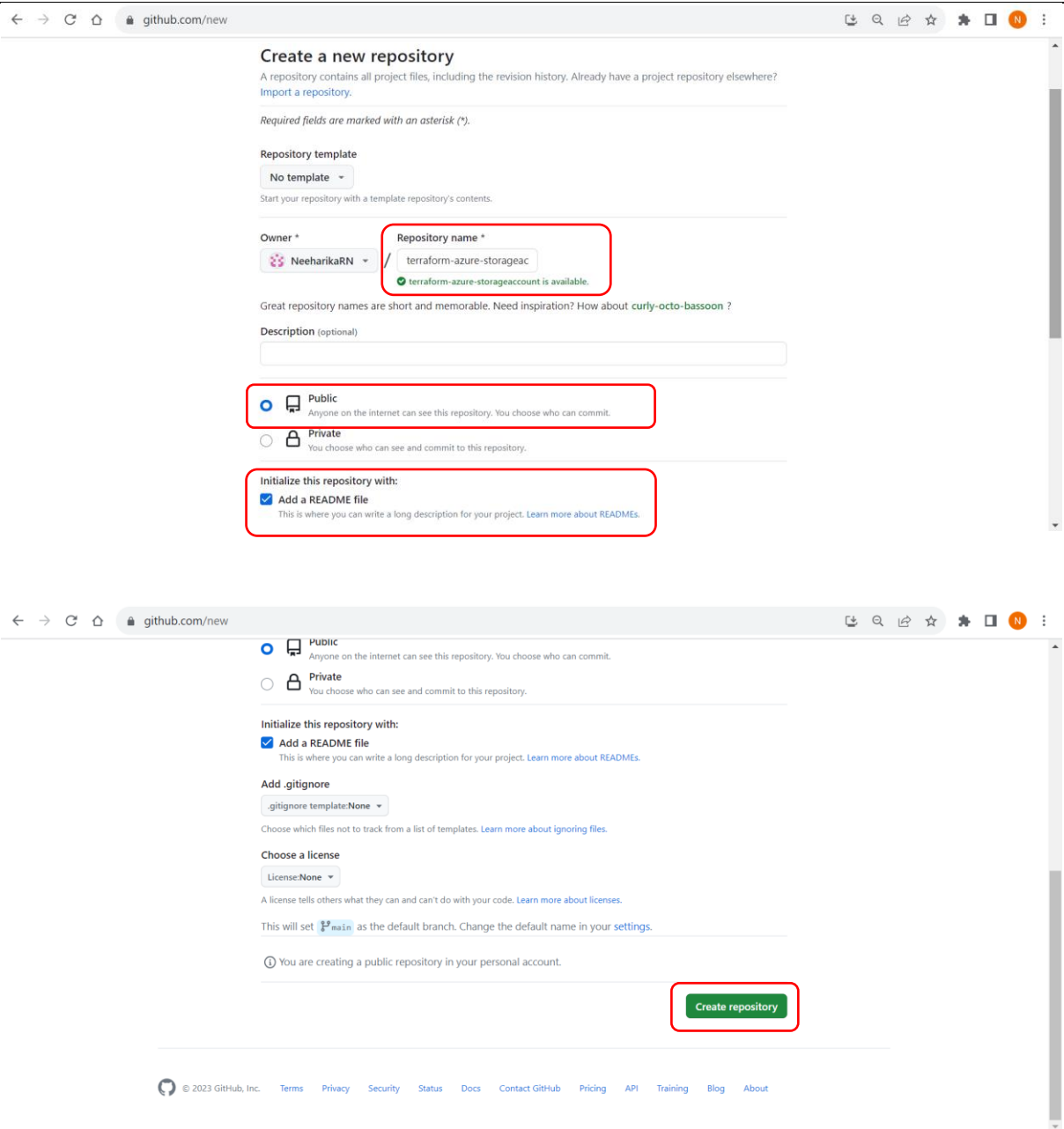
## Infrastructure Automation using Terraform on Azure – Lab Guide

	 <p>Then you will be redirected to home page.</p> 
--	---

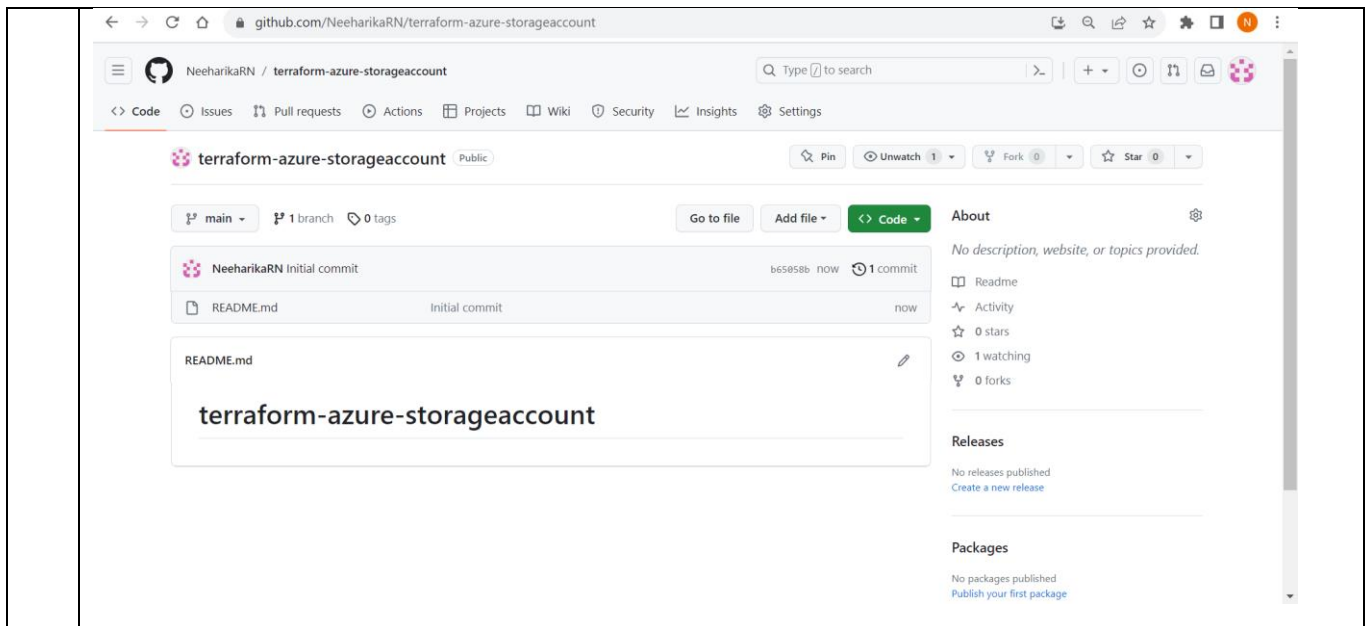
### Part 2: Create a new repository

1	<p>Click on Create repository or on “+” in your top right corner.</p> 
2	<p>Provide repository name as “terraform-&lt;provider_name&gt;-&lt;module_name&gt;” like “terraform-azure-storageaccount”, type of repository as Public, check the box “Add a Readme File”. Scroll Down and click on “Create Repository”.</p>



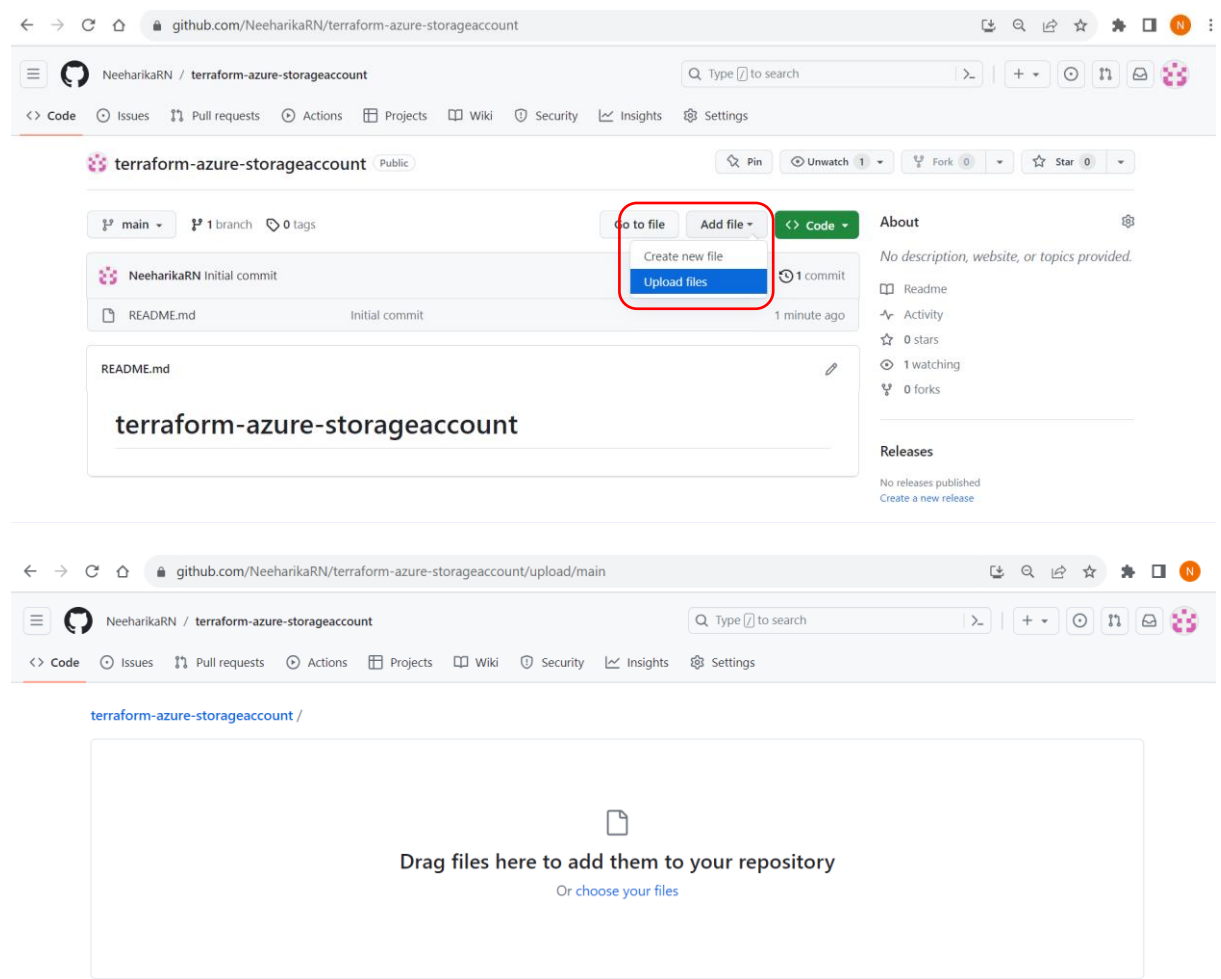
	 <p>The image displays two screenshots of the GitHub 'Create a new repository' page. The first screenshot shows the initial form with fields for Repository name, Owner, and Description. The second screenshot shows the same form with additional options like 'Add a README file', 'Add .gitignore', and 'Choose a license'.</p> <p><b>Repository name *</b> terraform-azure-storageaccount is available.</p> <p><b>Owner *</b> NeeharikaRN</p> <p><b>Repository template</b> No template</p> <p><b>Public</b> Anyone on the internet can see this repository. You choose who can commit.</p> <p><b>Private</b> You choose who can see and commit to this repository.</p> <p><b>Initialize this repository with:</b> <input checked="" type="checkbox"/> Add a README file This is where you can write a long description for your project. <a href="#">Learn more about READMEs.</a></p> <p><b>Add .gitignore</b> .gitignore template: None</p> <p><b>Choose a license</b> License: None</p> <p><b>Create repository</b></p>
3	Repository will be created and below page will be opened.

## Infrastructure Automation using Terraform on Azure – Lab Guide



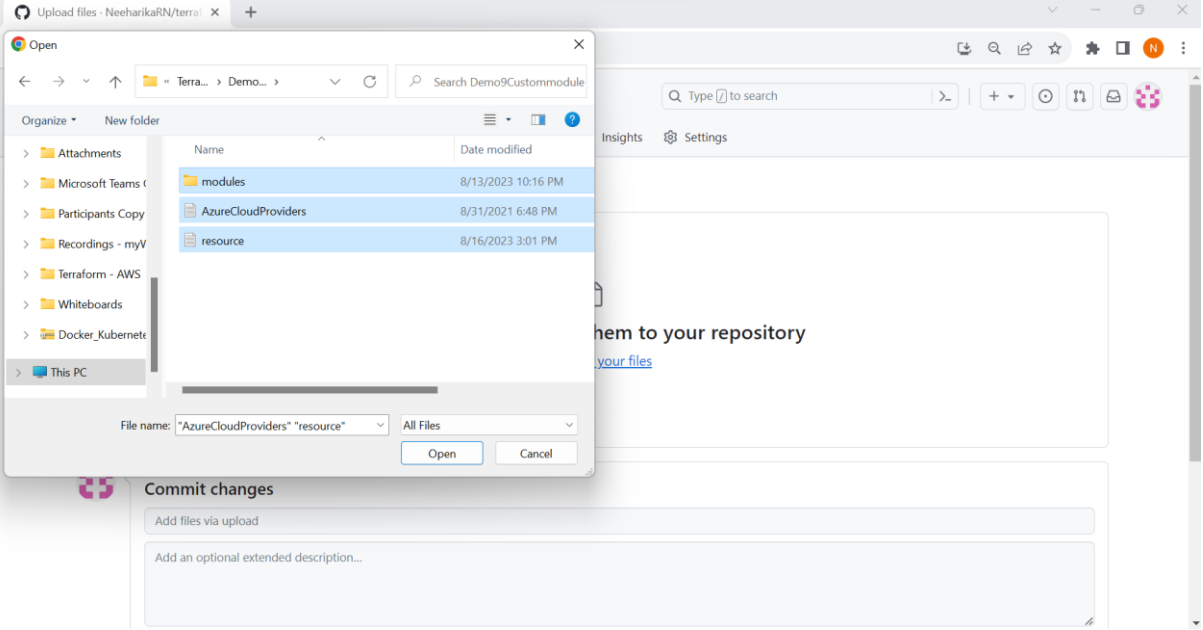
### Part 3: Upload files

- 1 Upload module files to above repository. Click on Add files and then choose Upload files.

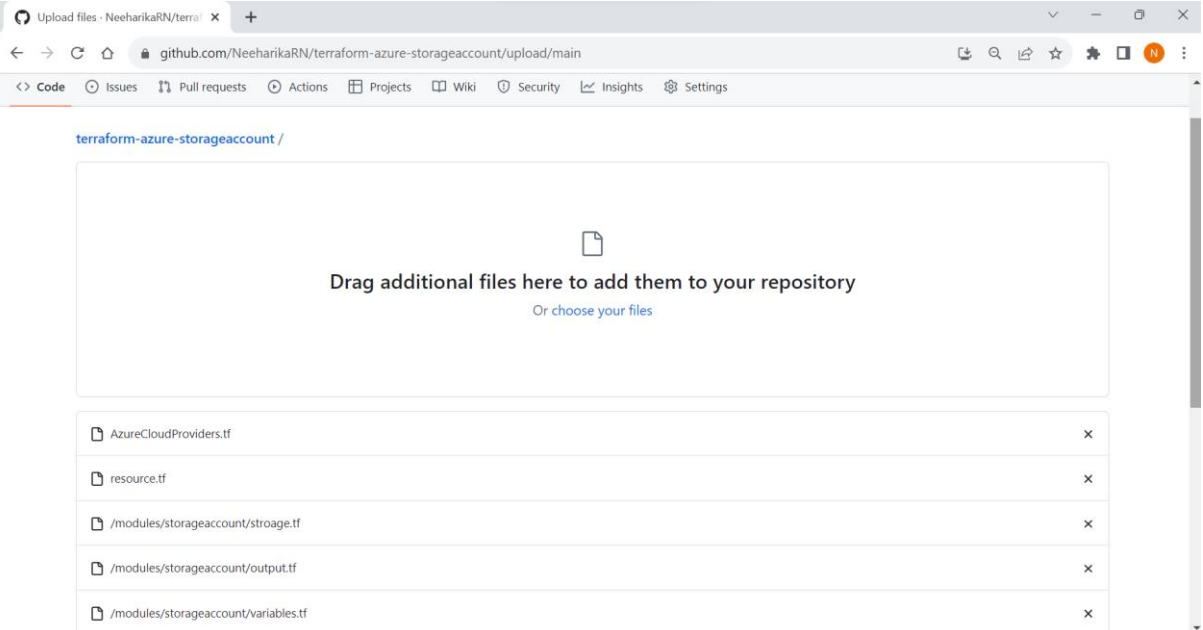


- 2 Drag and drop the required files or browse the files and click on upload.

## Infrastructure Automation using Terraform on Azure – Lab Guide



The screenshot shows a web browser window with the GitHub interface. An 'Open' dialog box is open, displaying the file structure of a repository. The 'modules' folder is selected, and the 'AzureCloudProviders' file is highlighted. The 'Commit changes' section is visible below the dialog.



The screenshot shows the GitHub repository page for 'terraform-azure-storageaccount'. The page displays the repository name and a list of files: 'AzureCloudProviders.tf', 'resource.tf', '/modules/storageaccount/stroage.tf', '/modules/storageaccount/output.tf', and '/modules/storageaccount/variables.tf'. The 'Commit changes' button is visible at the bottom.

3

Scroll down and click on commit changes.

## Infrastructure Automation using Terraform on Azure – Lab Guide

The top screenshot shows the GitHub 'Commit changes' dialog box. The 'Commit changes' button is highlighted with a red rectangle. The dialog box includes options to 'Commit directly to the main branch' or 'Create a new branch for this commit and start a pull request'. The 'Commit changes' button is located at the bottom left of the dialog box.

The bottom screenshot shows the GitHub repository page for 'terraform-azure-storageaccount'. The 'Releases' section is visible on the right side of the page, with a link to 'Create a new release'.

### Part 4: Create a new release

- 1 Click on "Create a new release" on the bottom right corner.

## Infrastructure Automation using Terraform on Azure – Lab Guide

The screenshot shows a GitHub repository page for 'terraform-azure-storageaccount' by user 'NeeharikaRN'. The repository is public and has 1 branch and 0 tags. The file list shows 'modules/storageaccount' (Add files via upload, now), 'AzureCloudProviders.tf' (Add files via upload, now), 'README.md' (Initial commit, 9 minutes ago), and 'resource.tf' (Add files via upload, now). The README section shows the repository name 'terraform-azure-storageaccount'. The right sidebar shows 'About' (No description, website, or topics provided), 'Releases' (No releases published, Create a new release), and 'Packages' (No packages published, Publish your first package).

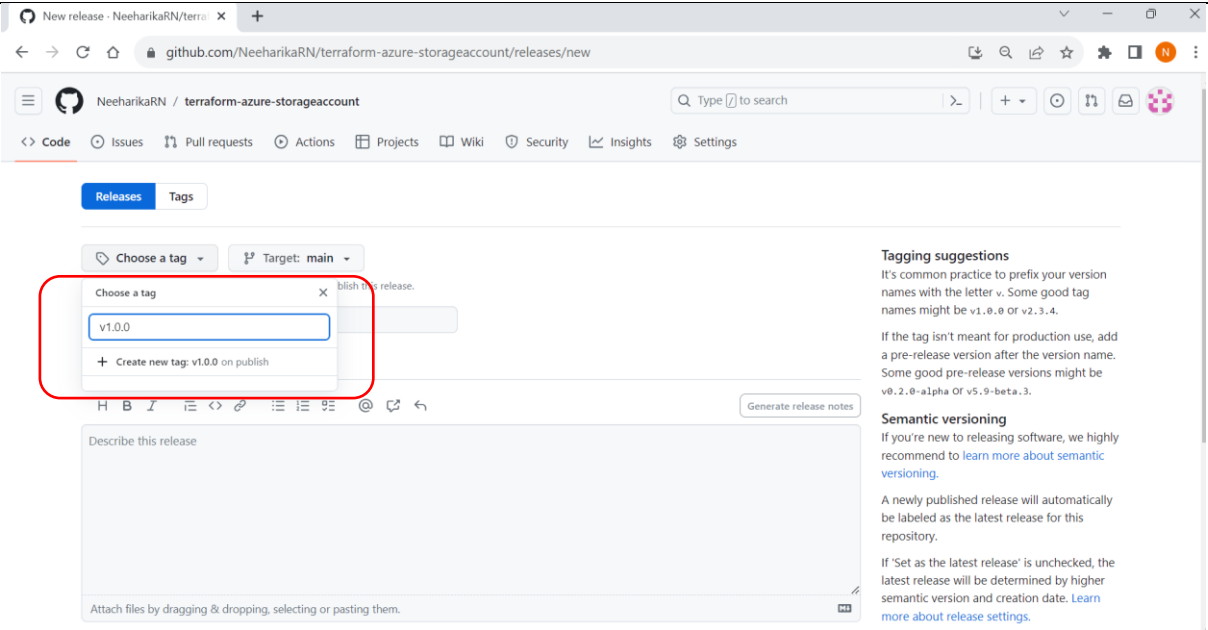
---

2

Click on "Choose a tag" . Enter tag as "v1.0.0" and click on "Create new tag".

The screenshot shows the GitHub interface for creating a new release. The repository is 'NeeharikaRN/terraform-azure-storageaccount'. The 'Releases' tab is active. A red box highlights the 'Choose a tag' dropdown menu, which is currently set to 'main'. Below the dropdown, a text input field for 'Release title' is visible. To the right, there are sections for 'Tagging suggestions' and 'Semantic versioning'.

## Infrastructure Automation using Terraform on Azure – Lab Guide



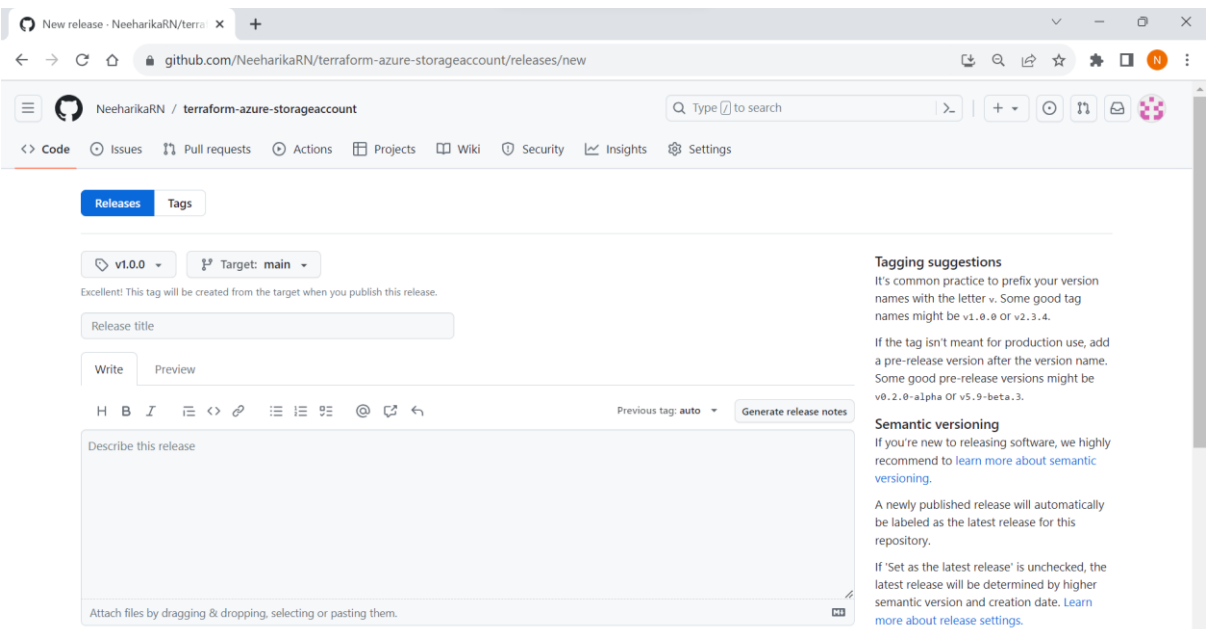
Choose a tag: v1.0.0

Target: main

Describe this release

Generate release notes

3 Scroll down and click on "Publish release".



Excellent! This tag will be created from the target when you publish this release.

Release title

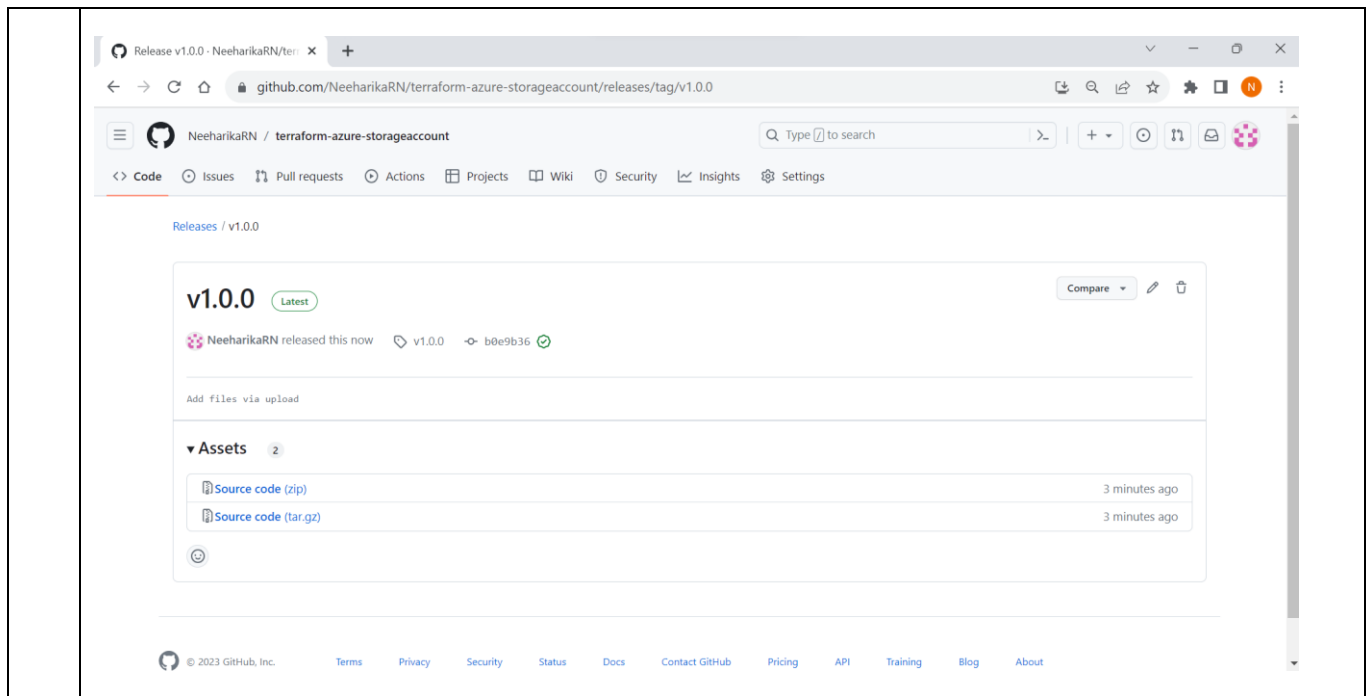
Describe this release

Generate release notes

☐ Set as a pre-release

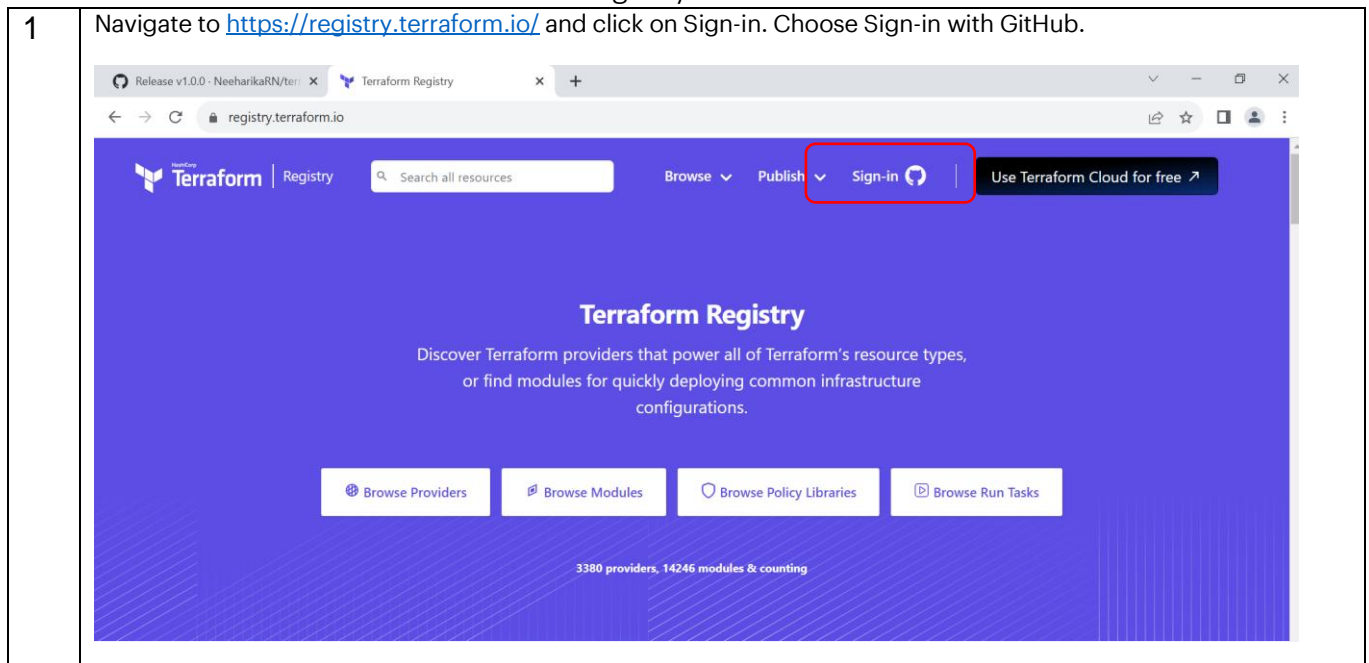
Publish release

## Infrastructure Automation using Terraform on Azure – Lab Guide



### Part 5: Publish the Module in Terraform Registry

1. Navigate to <https://registry.terraform.io/> and click on Sign-in. Choose Sign-in with GitHub.

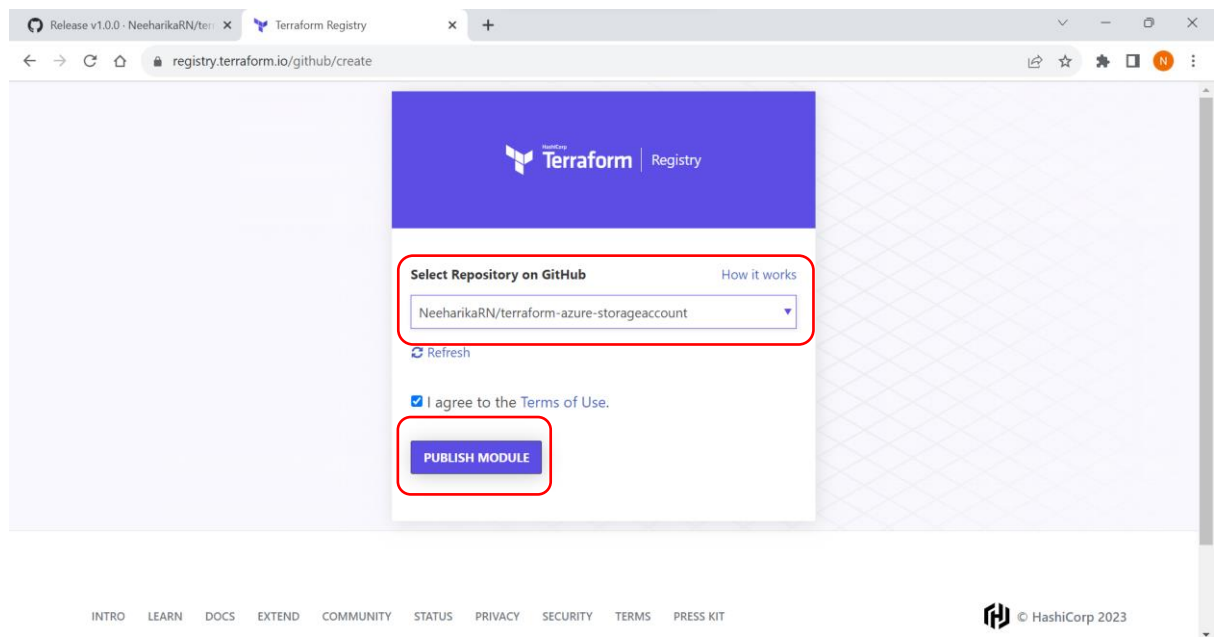


## Infrastructure Automation using Terraform on Azure – Lab Guide

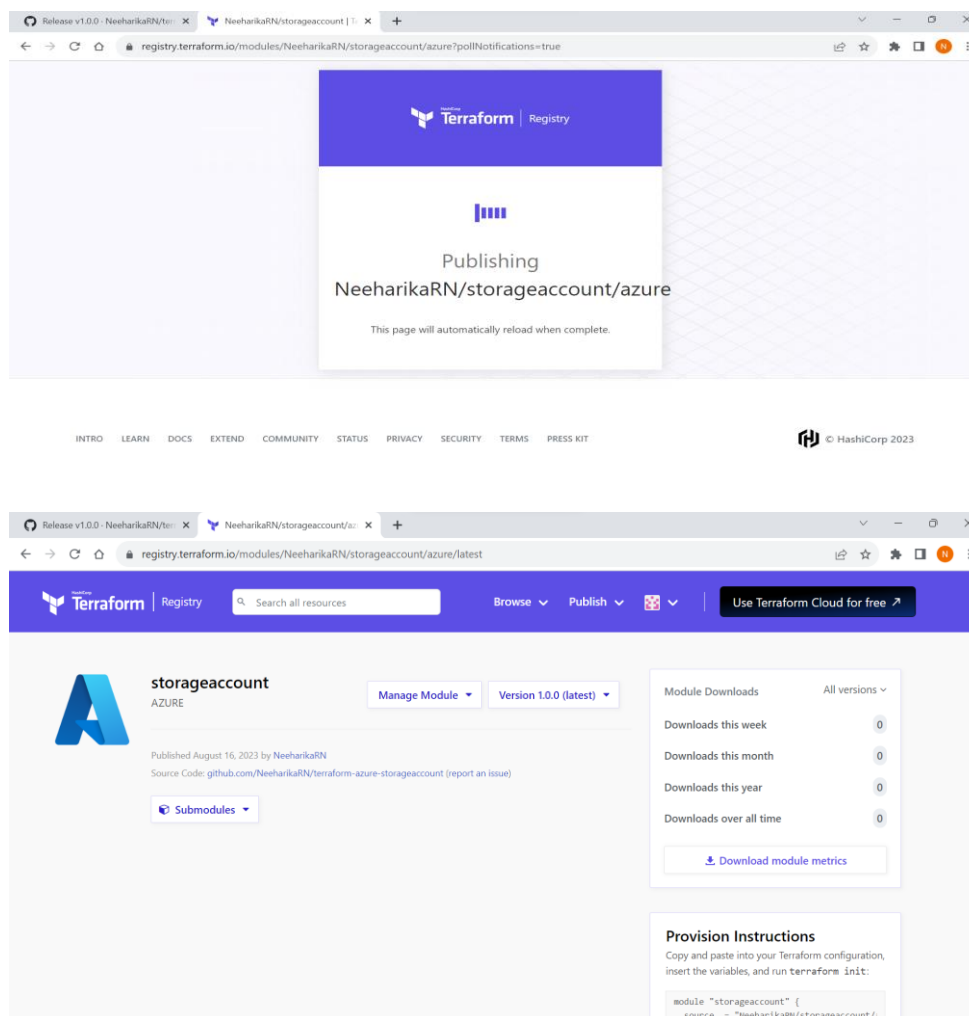
	
2	<p>Click on Publish and Choose Module.</p> 
3	<p>Click on the drop-down arrow and select the module which you want to publish.</p> 

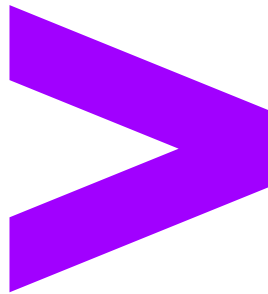


4 After selecting module , check the box “I agree to the Terms of Use” and Click on “Publish Module”.



5 After publishing module, you will be redirected to below page.





Copyright © 2024 Accenture  
All rights reserved.  
Accenture and its logo are trademarks of Accenture.